SIMULER EN ELECTROTECHNIQUE AVEC PSPICE

1 Présentation :

1.1 Public concerné :

Les enseignants en électrotechnique qui souhaitent intégrer la simulation dans leur pratique professionnelle. Sont concernés aussi bien les professeurs de physique appliquée que ceux de génie électrique.
L'utilisation de ce document suppose une connaissance préalable des fonctions de base de Pspice (des ressources sont proposées plus loin pour les débutants).

1.2 Pourquoi simuler?

Avec l'avènement de l'informatique pour tous les outils de simulation se sont répandus dans tous les secteurs d'activité. L'électrotechnique ne pouvait pas être épargnée. La simulation donne au technicien la possibilité de réaliser puis de tester un prototype virtuel. Les avantages sont évidents : économies de temps et d'argent.
Pour l'enseignant la simulation s'impose pour les raisons suivantes :

- Cette technique permet d'obtenir des résultats lorsque le matériel nécessaire manque : mesurer des courants, calculer la transformée de Fourier d'un signal.
- Les étudiants étant ce qu'ils sont et non ce que nous voudrions qu'ils soient, de nombreux phénomènes sont trop complexes pour une étude théorique, la simulation permet de montrer ce que l'on n'a pas le temps d'expliquer.
- La simulation est de plus en plus utilisée dans l'industrie, il faut donc y préparer nos étudiants.

Les limitations du procédé ne doivent pas être oubliées : si complexe soit-il, un modèle n'est jamais la réalité. C'est pourquoi la simulation ne remplacera jamais totalement l'expérimentation. A l'usage la simulation paraît un bon complément aux TP et aux expériences de cours.

1.3 Pourquoi Pspice ?

Nous avons choisi Pspice, produit par Microsim, pour les raisons suivantes :

- Pspice est le logiciel de simulation le plus répandu dans l'industrie.
- Il est basé sur le standard industriel SPICE et donne donc accès aux bibliothèques de modèles développées par les fabricants.
- Il permet de mélanger sans aucun problème des composants numériques et analogiques.
- La prise en main du logiciel est simple.
- C'est un logiciel très complet puisqu'il permet de simuler tout les aspects des systèmes que l'on rencontre en électrotechnique : électronique de puissance, électronique de commande analogique et numérique, asservissements, mécanique.
- Les limites de la version d'évaluation permettent son utilisation dans le cadre d'un lycée.

Au moins deux autres outils de simulation sont répandus : Simulink et Psim :

- Simulink est une extension de Matlab, un logiciel de mathématiques, destiné à la simulation des systèmes asservis. C'est un produit très puissant pour lequel des bibliothèques dédiées à l'électrotechnique ont été développées. Son inconvénient réside dans sa présentation très abstraite qui convient plutôt pour des universitaires ou des ingénieurs.

- Psim est, comme Pspice un simulateur qui utilise des composants. C'est un outil très complet et orienté vers l'électrotechnique. Ses défauts sont que seule la simulation temporelle est disponible et que l'outil de visualisation des résultats est très rudimentaire.

1.4 Quelle version utiliser ?

Le monde de l'informatique est en perpétuelle évolution. Au moment où ces lignes sont écrites la version 9.2 vient de sortir. Toutefois ce document a été réalisé avec la version 8. La différence essentielle entre la version 8 et la version 9 est le logiciel de saisie de schéma. Depuis la version 9 Microsim a été racheté par OrCad qui a intégré Pspice dans sa suite logicielle. Pspice est donc toujours disponible mais avec la saisie de schéma *Capture* (produite par OrCad) au lieu de la saisie de schéma *Schematics* (produite par Microsim). Ceci a provoqué une levée de bouclier des utilisateurs de Pspice qui voulaient continuer à utiliser la saisie de schéma *Schematics* qui est, de fait, bien supérieure à *Capture*. Le client est roi. En conséquence OrCad permet maintenant à ses utilisateurs de choisir au moment de l'installation du logiciel entre *Schematics* et *Capture*.
En bref :

- **Vous possédez Pspice version 8 :** Vous n'avez rien à modifier. Vous pouvez directement utiliser ce document.
- **Vous possédez Pspice version 9 :** Il faut l'installer en choisissant l'option *Schematics* (attention : cette option n'est pas disponible avec toutes les sous-versions d'OrCad V9).
- **Vous ne possédez pas Pspice :** Vous trouverez au **§2.1** des indications pour télécharger gratuitement Pspice**.**

1.5 Pourquoi la version d'évaluation ?

Comme beaucoup de fabricants de logiciel Microsim (maintenant OrCad) propose une version d'évaluation. Celle ci est entièrement gratuite, fonctionne sans clé, peut être installée sur autant de postes que l'on veut, être distribuée aux élèves et étudiants.
Toute rose ayant ses épines, cette version d'évaluation est limitée. Mais la limitation est uniquement sur le nombre de composants. Toutes les fonctionnalités du logiciel sont disponibles.
Les bibliothèques spécialisées et les méthodes présentées ici permettent de travailler avec la version d'évaluation de Pspice.

2 Les premiers pas :

2.1 Se procurer le logiciel :

Deux ouvrages récents ont été publiés avec un cdrom inclus contenant Pspice version 8 :

- Analyse et conception de circuits électriques avec Pspice/DesignLab
  Auteurs : Marc Bougeant et Thierry Royant
  Editeur : EYROLLES

Cet ouvrage est par ailleurs une excellente initiation à Pspice que nous recommandons aux débutants.

- La vitesse variable électrique
  Auteur : François Bernot
  Editeur : International Thomson Publishing

Le site ressource (voir §2.6) associé à ce document donne des explications et des liens mis à jour pour télécharger gratuitement le logiciel.

2.2 S'initier à Pspice :

Pour ceux qui n'ont jamais utilisé Pspice, un document d'initiation rédigé par un collègue, Serge MONIN, est disponible sur le site ressource (voir §2.6).

2.3 Les limitations de la version d'évaluation :

La version d'évaluation de Pspice présente les limitations suivantes :

- Une bibliothèque de modèles réduite : 39 en analogique et 134 en numérique.
- Schéma limité à 64 équipotentielles, ou 10 transistors ou 2 amplificateurs opérationnels ou 65 fonctions logiques.
- Possibilité de configurer une seule bibliothèque supplémentaire comprenant 20 composants.

Le premier point n'est pas trop gênant grâce à la possibilité de créer les composants dont on a besoin. D'autant que de nombreux fabricants mettent à la disposition des utilisateurs les modèles SPICE de leurs composants.

Le deuxième point mérite développement. Il existe deux sortes de composants sous Pspice :

- Les composants intrinsèques dont font partie les composants passifs, les sources, et certains composants actifs (diodes, transistors, IGBT).
- Les sous-circuits dont font partie les thyristors, les ampli-op et tous les circuits intégrés. Bien qu'apparaissant sous *Schematics* comme des symboles ordinaires ils sont vus par *Pspice* comme des ensembles de composants. C'est ainsi qu'avec une poignée de thyristors ou quelques ampli-op on peut dépasser les limites de la version d'évaluation.

Si l'on a compris que le problème vient des sous-circuits la solution apparaît. Il faut créer des modèles simplifiés des sous-circuits dont on a besoin. La bibliothèque jointe à ce document propose des modèles de thyristor, ampli-op, MCC conçus pour être économes en composants.

2.4 L'histoire de Pspice :

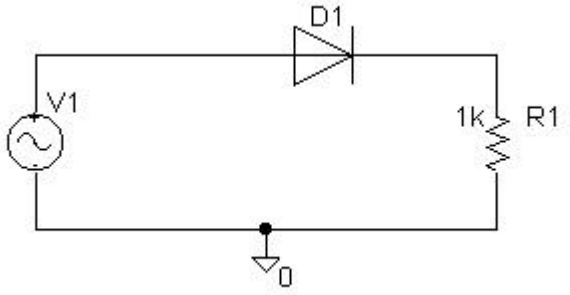
Connaître l'histoire de Pspice permet de comprendre quelques particularités de ce logiciel.

Dans les années 70, des étudiants en informatique de l'université de Berkeley en Californie mettent au point un algorithme destiné à la conception des circuits intégrés analogiques. Cet algorithme se nomme SPICE (***S**imulation **P**rogram with **I**ntegrated **C**ircuits **E**mphasis*). Par la suite différentes entreprises vont commercialiser des logiciels de simulation basés sur le noyau SPICE. L'évolution de l'informatique permettra de porter ces logiciels successivement sur station de travail puis, à la fin des années 80 sur micro-ordinateurs. La version de SPICE vendue par Microsim devient le standard de l'industrie sous le nom de Pspice.

Microsim a ajouté au noyau Spice originel la simulation numérique et la possibilité de mélanger circuits numériques et analogique.

Pspice a gardé de ses origines quelques caractéristiques :

- Pas de distinction entre majuscules et minuscules.
- Pas de caractères spéciaux (µ ou Ω).
- Le logiciel de simulation proprement dit (*Pspice*) ne fonctionne qu'en mode texte. Il faut donc une interface d'entrée (*Schematics*) qui converti le schéma en texte. Par exemple un schéma tel que celui ci :



Devient le fichier texte :

```
D_D1          $N_0001 $N_0002  D1N4007
R_R1          $N_0002 0  1k
V_V1          $N_0001 0 SIN 0 325 50 0 0 0
```

- L'algorithme SPICE ne peut converger que si tous les courants et tensions sont des fonctions continues du temps. Pour obtenir cette continuité toutes les diodes possèdent une capacité parasite (comme les diodes réelles). Lorsque l'on crée un circuit avec des inductances ceci peut provoquer des oscillations parasites.
- Toutes valeurs par défaut de paramètres tels que les valeurs max. des courants et tensions sont adaptées à la simulation de circuit intégrés analogiques c'est à dire ont des valeurs très faibles.

2.5 Le contenu du fichier *reselecspi.zip* :

Le fichier compressé qui est joint à ce document contient les fichiers suivants :

- enrea.lib : la bibliothèque de modèles.
- enrea.slb : la bibliothèque de symboles.
- Pd3m.sch et Pd3m.prb : les fichiers relatifs au thème n°1.
- mcc_ind.sch, mcc_ind.prb, mcc_freq.sch, mcc_freq.prb : les fichiers relatifs au thème n°2.
- ond_mono_pleinonde.sch, ond_mono_pleinonde.prb, ond_mono_MLI_sintri.sch, ond_mono_MLI_sintri.prb : les fichiers relatifs au thème n°3.
- pspice_resume.pdf : un mode d'emploi condensé sur deux pages de Pspice.
- pspice_TP1.pdf : un TP d'initiation à Pspice sur le redressement monophasé.
- pspice_TP2.pdf : un TP d'initiation à Pspice sur le redressement triphasé associé à une MCC.

2.6 Ressources en ligne :

J'ai créé sur le site de la STS Electrotechnique du Lycée Newton-ENREA de Clichy une page consacrée à la mise à jour des informations contenues dans ce document (adresse : http://membres.tripod.fr/tsetclichy/reselec.html).
On y trouvera des ressources sur :

- Le téléchargement de Pspice
- Bibliothèques pour les version 6 et 9
- Exemples de simulation
- Liens divers

2.7 Installer les bibliothèques spécialisées :

Ce document est accompagné d'un fichier (reselecspi.zip) qui contient les bibliothèques qui permettent de réaliser les simulations proposées. On y trouvera le fichier *enrea.lib* qui est la bibliothèque de modèles et *enrea.slb* qui est la bibliothèque de symboles. Il faut configurer Pspice pour ces bibliothèques :

- Installer winzip si ce n'est déjà fait (ce logiciel est disponible gratuitement en téléchargement à l'adresse : http://www.winzip.com/winzip.htm)
- Double cliquer sur le fichier reselec.zip
- Faire *Action* *Extract*
- Sélectionner le répertoire **C:**
- Cocher la case *Use Folder Names*
- Cliquer sur *Extract*

- A ce stade vous devez trouver les fichiers *enrea.lib* (fichier contenant les modèles des composants) et *enrea.slb* (contenant les symboles des composants) dans le répertoire *c:\msimev_8\userlib*. Les autres fichiers doivent être dans le répertoire *c:\msimev_8/projects/reselec*.
- lancer *Schematics*
- Cliquer sur *Analysis* *Library and include files* *Browse*
- Ouvrir le sous-répertoire *userlib* et sélectionner le fichier *enrea.lib*
- Cliquer sur *OK* *Add Library** *OK*
- Cliquer sur *Options* *Editor configuration* *Library settings* *Browse*
- Ouvrir le sous-répertoire *userlib* et sélectionner le fichier *enrea.slb*
- Cliquer sur *OK* *Add ** *OK*

2.8 Adapter Pspice à l'électronique de puissance :

Lorsque l'on simule des montages d'électronique de puissance avec Pspice on est souvent confronté à un blocage accompagné du message énigmatique :

```
ERROR     Discontinuing simulation due to convergence problem
```

Ce message signifie que l'algorithme de résolution du circuit n'arrive pas à converger. Ceci vient du fait que l'algorithme de Pspice est en double précision sur un format de 15 bits. Il en résulte les limites suivantes :

- Les courant et les tensions ne peuvent pas dépasser $\pm 1.10^{10}$ V ou A.
- Les dérivées sont limitées à 1.10^{14} .
- En simulation temporelle le pas de calcul (*time step)* minimum dépend de la durée de la simulation.

On trouvera dans la documentation de Pspice une discussion complète sur ce sujet (voir Pspice A/D User's guide : Appendix B Convergence and Time Step Too Small Errors).
Dans la pratique, les ennuis arrivent lorsque l'on travaille avec des tensions ou des courants élevés ou lorsque la durée de la simulation est grande devant la période des phénomènes les plus rapides. Les solutions à mettre en oeuvre sont les suivantes :

- Tout d'abord vérifier le schéma. Une erreur de conception entraînant des valeurs extrêmes pour les courants ou les tensions se traduit par une non-convergence. Un exemple classique est l'utilisation d'une source de courant lorsqu'il n'y a pas de chemin pour le courant (résultat : une tension qui tend vers l'infini).
- Modifier les paramètres suivants (par *Analysis* *Setup* *Options*) : Abstol = 1nA
  Reltol = 0.01 Vntol = 1mV
- Remplacer les inductances pure par des inductances à noyau de fer (voir thème n°1).
- Diminuer la durée de la simulation.
- Placer la référence de tension (composant *AGND)* à un autre endroit.
- Allonger la durée des fronts sur les sources *Vpulse* (fixer TD et TF ≠ 0).

2. 9 Mode d'emploi du document :

Ce texte est une auto formation autour de quatre thèmes de l'électrotechnique :

- Redressement triphasé
- Machine à courant continu
- Onduleur monophasée
- Source triphasée

Le lecteur est appelé à suivre les instructions pour réaliser les simulations qui lui sont proposées. Un double encadré signale les parties purement informatives.

THEME N°1 : LE REDRESSEMENT COMMANDE

Nous allons réaliser la simulation d'un pont mixte triphasé. Le pont débitera d'abord sur une charge RL puis nous remplacerons celle ci par un générateur de courant afin de tracer la caractéristique $\overline{U}_{CH} = f(\alpha)$.

Placer une source triphasée 400V : On prendra trois sources de tension *vsin* reliées en étoile (attention aux polarités des sources). Les paramétrer comme suit :

- Paramètres communs aux trois sources : VOFF = 0 , VAMPL = 325V , FREQ = 50Hz
- Pour *V1* : PHASE = 0
- Pour *V2* : PHASE = 240
- Pour *V3* : PHASE = 120

Le paramètre PHASE correspond à la phase en degrés à l'instant t = 0 . Les valeurs ci-dessus donnent un système triphasé direct.

Placer les diodes :

- Choisir des D1N4007 dans la bibliothèque ENREA.

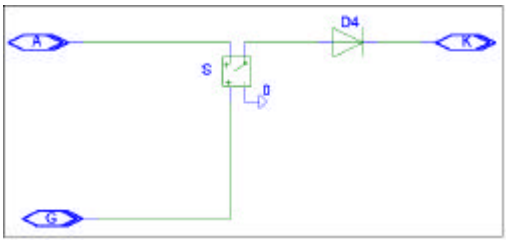
Placer les thyristors :

- Choisir des *THtri* dans la bibliothèque ENREA. Les paramétrer comme suit.
- Pour TH1 : ALPHA = {retard + 30}
- Pour TH2 : ALPHA = {retard - 210}
- Pour TH3 : ALPHA = {retard - 90}

Définir le retard à l'amorçage :

- Placer un composant *param* de la bibliothèque SPECIAL.
- Le paramétrer comme suit : NAME1 = retard VALUE1 = 45

La version d'évaluation de Pspice propose un thyristor 2N1595 de Motorola. Ce modèle a un inconvénient grave : il suffit d'en placer plus de trois sur un schéma pour dépasser les limitations de la version d'évaluation. Pourquoi ? Parce que ce composant est un sous-circuit composé d'un grand nombre de composants intrinsèques de Pspice. La solution ? Créer un modèle simplifié de thyristor composé de quelques composants intrinsèques. Le modèle de thyristor *THtri* a été conçu pour fonctionner dans le cadre des montages redresseurs triphasés sur réseau 50Hz. Voici le schéma du sous-circuit correspondant :



On reconnaît un interrupteur commandé en série avec une diode. L'interrupteur se ferme à l'instant *alpha* exprimé en degrés et s'ouvre 120° plus tard.

Pour réaliser la simulation de montages redresseurs monophasés ou de montages gradateurs on utilisera le composant *THmono* dont la durée de conduction est de 180°.

Utilisation de *THtri*

Le paramètre *alpha* associé à chaque thyristor est l'instant de mise en conduction exprimé en degrés de ce thyristor. Pour un pont mixte triphasé et un retard à l'amorçage de 0° on aura donc :

- *alpha* = 30° pour le thyristor TH1 (rappel : V_1 est une fonction sinus du temps)
- *alpha* = 30° + 120° = 150° pour le thyristor TH2
- *alpha* = 30° + 240° = 270° pour le thyristor TH3

Pour modifier le retard à l'amorçage il faut recalculer les angles et les modifier pour chaque thyristor. Ceci est fastidieux. On va donc créer un paramètre qui correspond au retard à l'amorçage du pont. Pour cela il faut créer le paramètre *retard* à l'aide du composant *param*. On obtient :

- *alpha* = {retard + 30°} pour le thyristor TH1
- *alpha* = {retard + 150°} pour le thyristor TH2
- *alpha* = {retard + 270°} pour le thyristor TH3

Ceci fonctionne de la manière suivante : lors de la compilation du schéma le logiciel calcule l'expression entre crochet en prenant pour valeur de *retard* celle qui est affichée sous le composant *param*. On peut ainsi modifier très simplement la valeur du retard à l'amorçage.

Pour finir : il est souhaitable qu'à l'instant t = 0 au moins un des thyristors conduise. Ceci est obtenu en retranchant 360° à la valeur de *alpha* pour les thyristors TH2 et TH3. On écrira donc finalement :

- *alpha* = {retard + 30°} pour le thyristor TH1
- *alpha* = {retard - 210°} pour le thyristor TH2
- *alpha* = {retard - 90°} pour le thyristor TH3

La charge : Placer une résistance de 22 Ω et une inductance de 30 mH (composant *Lenrea* de la bibliothèque ENREA).

Dans les montages d'électronique de puissance il est préférable d'éviter d'utiliser l'inductance proposée par Pspice. En effet celle ci est une inductance parfaite. Cette inductance peut créer avec les capacités parasites des diodes des oscillations à très haute fréquence. Ces oscillations sont totalement irréalistes puisque les inductances réelles que l'on utilise en électronique de puissance sont des inductances à noyau de fer. Les pertes fer augmentent avec la fréquence ce qui revient à dire qu'en haute fréquence ces inductances se comportent comme des résistances.

Par ailleurs ces oscillations ralentissent voir bloquent la simulation, provoquant les fameux messages « *convergence problem* ».

La solution consiste à ajouter une résistance en parallèle avec l'inductance parfaite qui simule les pertes fer et évite les oscillations. Le composant *Lenrea* correspond à ce montage. La valeur par défaut de la résistance est 1kΩ. Elle peut être modifiée.

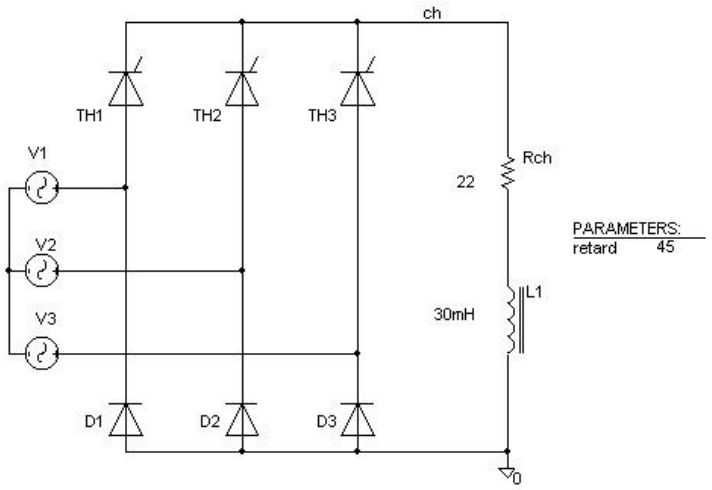
La référence de potentiel : Ne pas oublier cet élément indispensable. Placer un composant *AGND* sur le potentiel des anodes communes des diodes.

Préparer la visualisation des résultats :

- Changer le nom de la résistance en Rch.
- Placer une étiquette *ch* sur la sortie du pont en double-cliquant sur la connexion correspondante.

Il est utile de prévoir dès la saisie du schéma une visualisation aisée des résultat de la simulation. Pour cela il faut donner des noms explicites aux composants (pour les courants), mettre des étiquettes (pour les tensions) et placer judicieusement la référence de potentiel. Dans notre exemple ceci nous permettra d'afficher V(ch) pour la tension aux bornes de la charge et I(Rch) pour le courant dans celle ci.

On obtient :



Paramètres de la simulation : On effectue une simulation temporelle (*transient)* paramétrée comme suit :

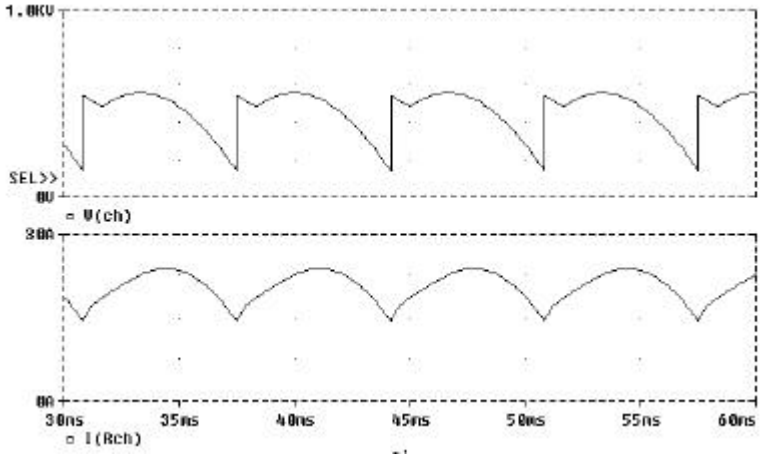
- *print step* = 20ns
- *final time* = 60ms
- *step ceiling* = 0.1ms

Le paramètre *print step* ne doit généralement pas être modifié. *Step ceiling* défini la valeur maximum du pas de calcul. Si les courbes obtenues ont bonne allure il est préférable de ne pas y toucher. Si elles sont anguleuses on peut le régler à une valeur qui sera suffisamment faible pour lisser les courbes mais pas trop pour ne pas ralentir excessivement la simulation. La valeur proposée convient pour les montages redresseurs.

Résultats de la simulation : Lancer la simulation. On peut alors visualiser :

- La tension aux bornes de la charge en affichant V(ch)
- Le courant dans la charge en affichant I(Rch)

On obtient :

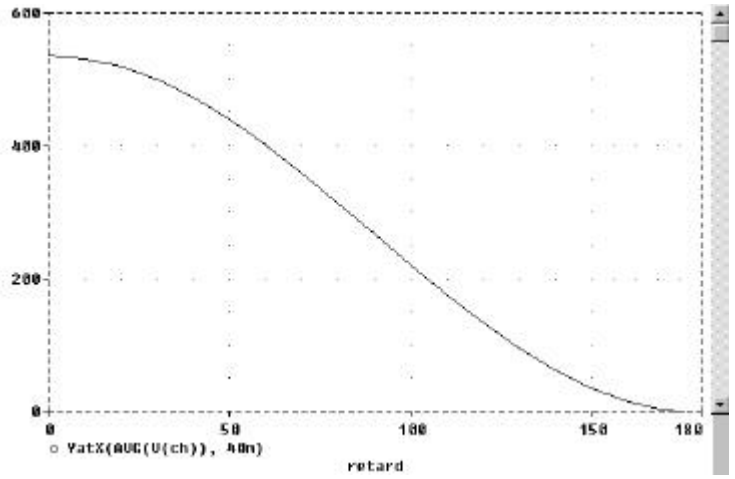


Si l'on visualise le courant dans un semi-conducteur ou dans une des sources de tension on observera des pics de courant d'intensité très grande et de durée très brève. Ceci est dû à la nature même de Pspice qui est un logiciel conçu pour la simulation des circuits linéaires. On peut supprimer ces pics de courant parfaitement irréalistes en plaçant en série avec les sources des inductances simulant l'impédance de celles ci. Mais on aura alors des pics sur les tensions…

Tracer la caractéristique $\overline{U}_{CH} = f(\alpha)$:

L'objectif est de tracer les variations de la valeur moyenne de la tension aux bornes de la charge en fonction du retard à l'amorçage. Pour cela nous allons effectuer une simulation multiple en faisant varier le paramètre *retard* puis utiliser une *goal function* dans Probe.

- Définir une simulation multiple : *Analysis* *Setup* *Parametric*
- Définir le paramètre à faire varier : sélectionner *Global Parameter* puis dans la case *Name* écrire *retard*
- Choisir une variation linéaire : sélectionner *Linear*
- Indiquer la valeur de départ, la valeur finale et l'incrément : *Start Value* = 0
  End Value = 180 *Increment* = 10
- Remplacer la charge RL par un générateur de courant (composant *IDC)* réglé à 1A.
- Lancer la simulation.
- Au démarrage de Probe celui ci propose par défaut les données correspondant à toutes les valeurs de *retard*. Cliquer sur *OK*.
- Nous allons maintenant tracer $\overline{U}_{CH} = f(\alpha)$: cliquer sur *Trace*
  Performance Analysis puis choisir l'assistant en cliquant sur *Wizard* .
- L'étape 1 est une explication du processus : cliquer sur *Next* .
- L'étape 2 est le choix de la *Goal Function* : choisir *YatX* (c'est une fonction qui renvoie la valeur d'une trace à un instant donné) puis cliquer sur *Next* .
- L'étape 3 est le choix de la trace à traiter. Dans la case *Name of trace* écrire *AVG(V(ch))* (on peut aussi cliquer sur l'icône placé à côté). Dans la case *X value to get Y value at* écrire *40ms*. Cliquer sur *Next* .
- L'étape 4 affiche la valeur de la *Goal Function* pour une valeur du paramètre à des fins de vérification. Cliquer sur *Next* .
- On obtient :



Ce qui correspond bien sûr au résultat attendu.

THEME N°2 : LA MACHINE A COURANT CONTINU

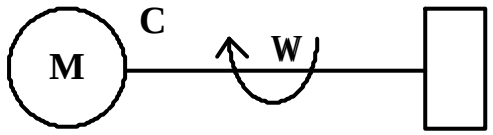
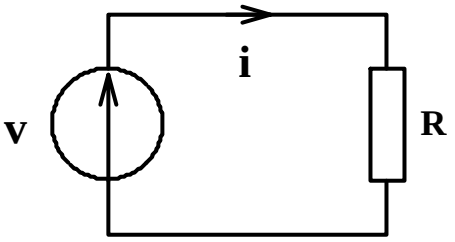
A travers la simulation d'une machine à courant continu nous allons apprendre à traiter les grandeurs mécaniques sous Pspice.

La simulation des grandeurs mécaniques :

SPICE a été conçu pour simuler des circuits intégrés analogiques. Il ne possède donc bien évidemment pas de la possibilité de simuler les grandeurs mécaniques. Il est cependant possible de le faire en appliquant une analogie entre grandeurs mécaniques et grandeurs électriques. La correspondance choisie est :

vitesse angulaire	**Û**	**tension**
couple	**Û**	**courant**

Les unités étant celles du SI (couple en m.N et vitesse en rad/s).
Il en résulte les correspondances suivantes :

ELEMENT MECANIQUE	ELEMENT ELECTRIQUE
Un moteur et une charge reliés par un axe rigide (Ω est commun) : 	Dipôles en parallèle (v est commun) 
Moment d'inertie : $J\frac{d\Omega}{dt}=\sum C$	Condensateur : $C\frac{dv}{dt}=i$
Couple visqueux : $C=k.\Omega$	Conductance : $i=\frac{1}{R}v$

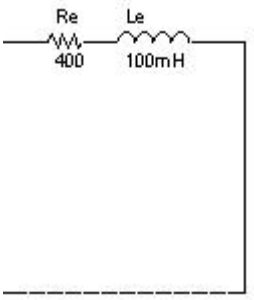
On pourra ainsi représenter les systèmes mécaniques en remplaçant chaque élément par son équivalent électrique.

Modélisation d'une machine à courant continu :

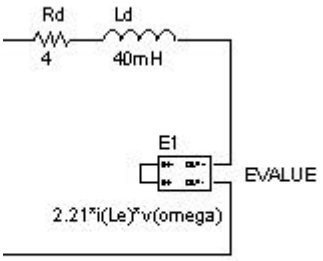
Nous allons appliquer ces notions à la modélisation d'une machine à courant continu de type Ecodime dont voici les caractéristiques :

Excitation : U = 220V R_e = 400Ω L_e = 100mH
Induit : U = 220V I = 7,2A R_d = 4Ω L_d = 40mH
P = 1,2kW N = 1500tr.mn^{-1}

La charge sera un volant d'inertie (J = 0.2 kg.m²) associé à un couple résistant proportionnel à la vitesse ($k = 46.10^{-3}$ m.N/tr.mn^{-1}).
Nous utiliserons les relations classiques $e = k.\Phi.\Omega$ et $C = k.\Phi.i_d$
Le modèle de l'excitation est très simple : une résistance en série avec une inductance :



Le modèle de l'induit contient en plus de la résistance et de l'inductance une fem proportionnelle au flux d'excitation et à la vitesse. A la place du flux d'excitation nous utiliserons le courant d'excitation ce qui revient à supposer qu'il n'y a pas de saturation. En ce qui concerne la vitesse nous avons vu qu'elle est représentée par une tension. L'expression de la fem sera donc : $e = K \cdot I_e \cdot v(\Omega)$ avec $K = \frac{(k\Phi)_{nom}}{I_{ex_nom}} = 2{,}21$. Pour modéliser la fem sous Pspice on utilisera un module de la bibliothèque ABM *(Analog Behavioral Modelling*) de type *Evalue*:



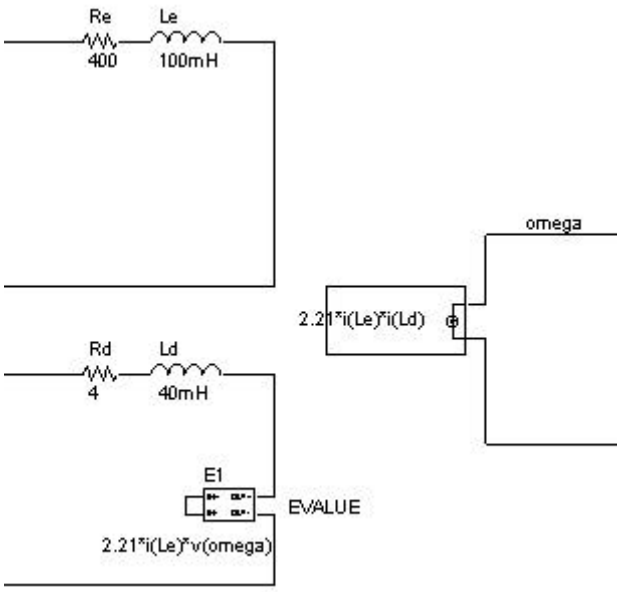
Le courant d'excitation est représenté par le courant dans l'inductance Le, la vitesse est représentée par le potentiel du point omega (voir plus loin) , l'expression de la fem sera donc, avec la syntaxe propre à Pspice :

2.21*i(Le)*v(omega)

Le couple de la MCC est donné par la relation $C = k.\Phi.i_d$. Nous utiliserons donc un générateur de courant de la bibliothèque ABM (composant *ABM/I*). L'expression du courant sera :

2.21*i(Le)*i(Ld)

i(Le) représente l'excitation, i(Ld) représente le courant d'induit. Le schéma de la MCC est maintenant complet :

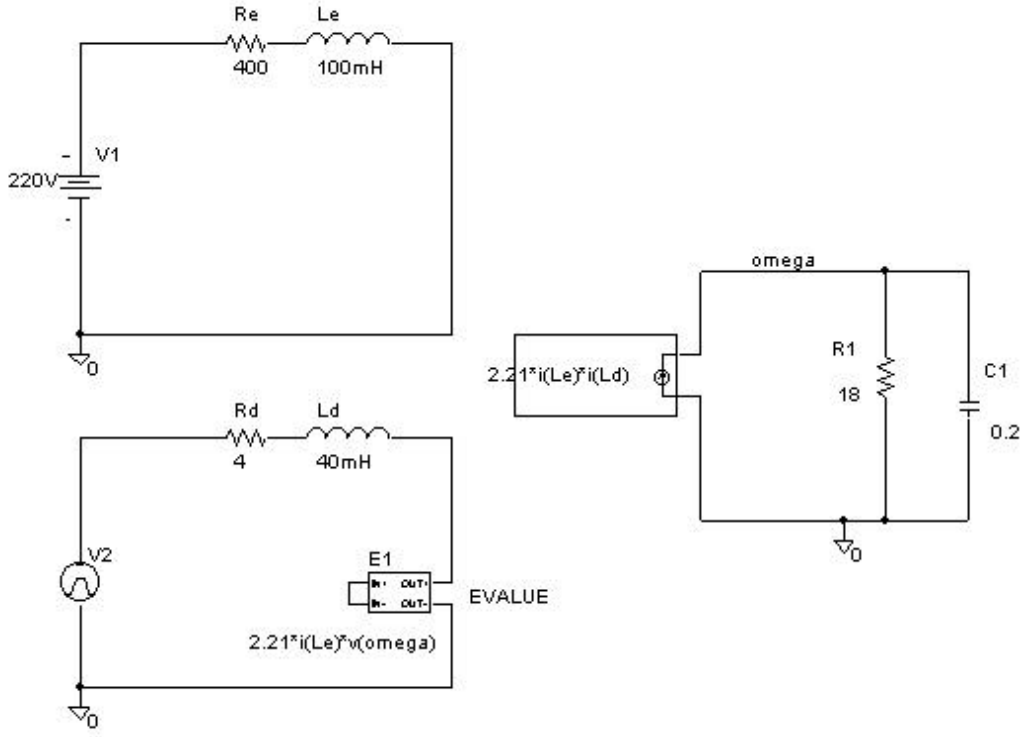


Essais du modèles de la MCC :

Nous allons à présent vérifier notre modèle en observant la réponse indicielle puis fréquentielle de celle ci.

- Alimenter l'inducteur avec une source de tension continue *VDC* réglée à 220V.
- Simuler la charge avec un condensateur de 0,2 F (pour l'inertie) en parallèle avec une résistance de 18 Ω (pour le couple résistant proportionnel à la vitesse).

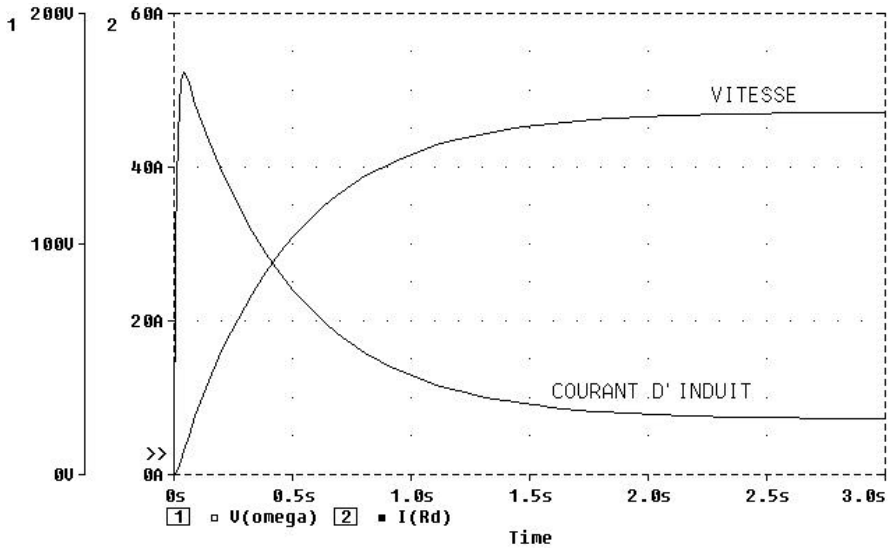
Réponse indicielle : On applique un échelon de tension d'amplitude 220 V. Ceci s'obtient avec une source de tension *Vpulse* (paramètres : V1 = 0 , V2 = 220V , les autres inchangés). Le schéma obtenu est le suivant :



- Configurer une simulation temporelle de durée 3s.

- Lancer la simulation.
- Relever simultanément la vitesse V(omega) et le courant dans l'induit I(Rd).

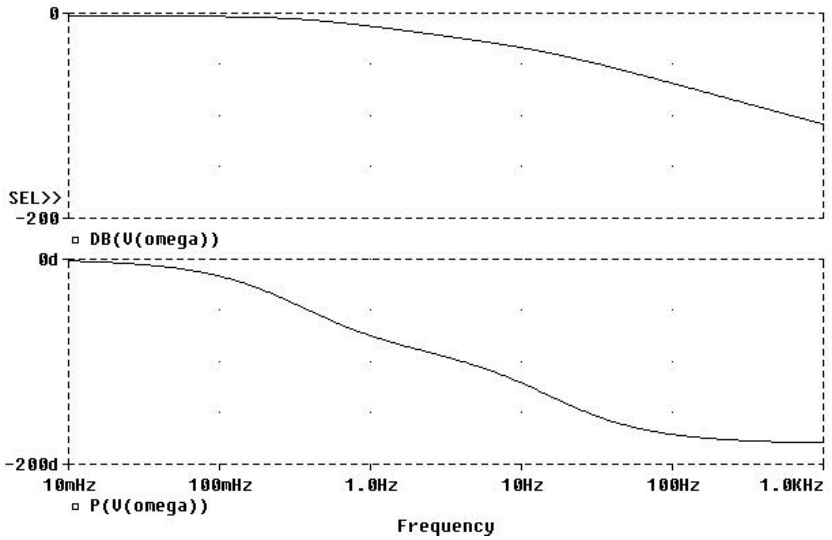
On obtient :



On observe bien la pointe d'intensité au démarrage, la montée progressive en vitesse et, en régime établi, une vitesse de 157 rd/s pour un courant d'induit de 7,2A.

Réponse en fréquence : Nous allons tracer la réponse en fréquence de la MCC considérée comme un système dont l'entrée est la tension d'induit et la sortie la vitesse angulaire.

- Remplacer la source *Vpulse* par une source *VAC* d'amplitude 1V (cette valeur permet de simplifier l'expression du gain).
- Sélectionner une simulation de type fréquentielle (*AC Sweep)*.
- Type de balayage : *Decade*
- Nombre de points de calcul : *Pts/decade* = 100
- Fréquence minimum : *Start Freq* = 0.01
- Fréquence maximum : *End Freq* = 1k
- Lancer la simulation.
- Tracer la phase : P(V(omega))
- Ajouter un écran : *Plot Add Plot*
- Tracer le gain : DB(V(omega))
- On obtient :



Le résultat est conforme à nos attente : un système du deuxième ordre amorti.

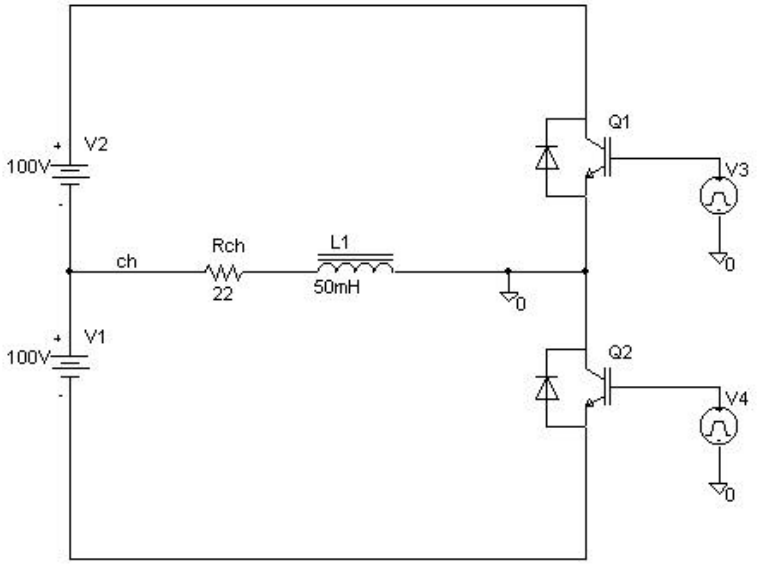
THEME N°3 : ONDULEUR MONOPHASE

Nous allons simuler un onduleur monophasé. Nous utiliserons d'abord une commande pleine onde puis une commande MLI sinus-triangle. Ce thème nous permettra d'aborder la mise en oeuvre des IGBT simplifiés de la bibliothèque ENREA, celle des EPROM ainsi que l'analyse harmonique.

Onduleur monophasé pleine onde :

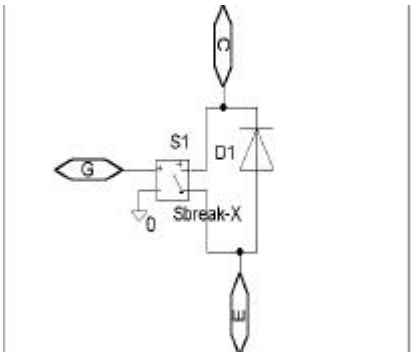
On choisit une structure en demi-pont :

- Placer deux sources de tension continues *VDC* réglées à 100V.
- Placer deux composants IGBT_D de la bibliothèque ENREA.
- Placer une charge RL de valeur 22 Ω, 50mH (prendre une inductance Lenrea).
- Placer deux générateurs de tension *Vpulse* pour la commande des IGBT.
- Placer une référence de potentiel *AGND* et une étiquette *ch*, nommer la résistance Rch.
- On obtient :



Mise en oeuvre des IGBT de la bibliothèque ENREA :

Le composant IGBT_D de la bibliothèque ENREA est constitué d'un interrupteur commandé en tension avec une diode en antiparallèle :



Cette structure très simple permet, avec deux composants seulement, de simuler un IGBT avec diode en antiparallèle et son interface de commande. Pour le commander il suffit d'appliquer ou non une tension de 12V sur la grille. On notera qu'il y a un isolement galvanique entre la pseudo grille et le circuit émetteur-collecteur.

Pour commander les IGBT on applique des créneaux d'amplitude 12V, de fréquence 50Hz, de rapport cyclique ½, dont l'un est en retard de $\pi/2$. On paramètre les sources *Vpulse* comme suit :

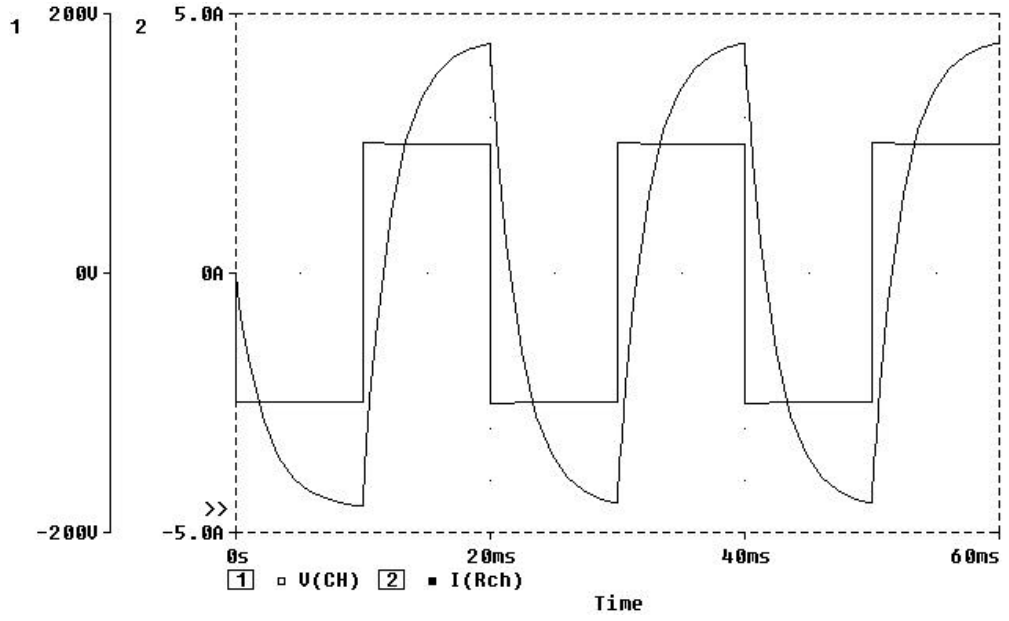
Paramètres communs aux deux sources *Vpulse* :

Tension basse : V1 = 0
Tension haute : V2 = 12V
Temps de montée : TR = 0
Temps de descente : TF = 0
Largeur de l'impulsion : PW = 10ms
Période : PER = 20ms

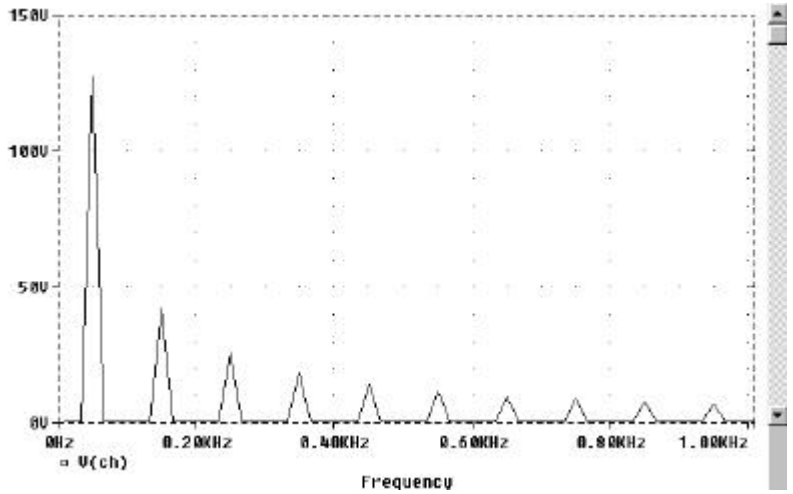
Paramètre propre à chaque source *Vpulse* :

Temps de retard : TD = 0 pour une source TD = 10ms pour l'autre

- Choisir une simulation temporelle de 60ms .
- Lancer la simulation.
- Afficher la tension aux bornes de la charge V(ch) et le courant dans la charge I(Rch).
- On obtient :



A l'aide de la commande *Trace Fourier* on peut faire l'analyse harmonique de la tension de sortie :

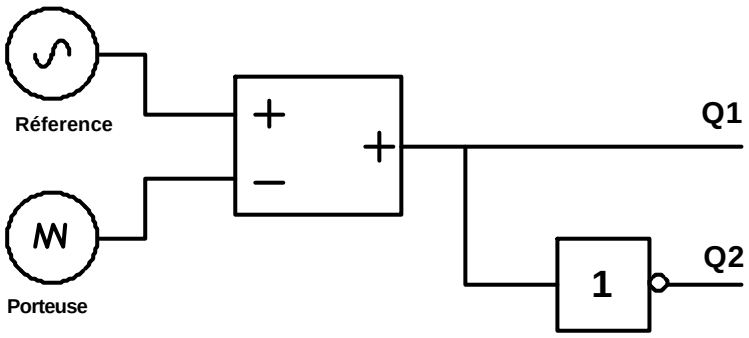


La valeur efficace du fondamental est V_1 = 90,2 V et l'on vérifie que tous les harmoniques impairs sont présents.

Onduleur monophasé à commande MLI sinus triangle :

L'objectif de cette commande est de repousser les premiers harmoniques à des fréquences élevées afin de faciliter le filtrage. On gardera le même schéma de puissance que pour l'onduleur pleine onde ce qui permettra les comparaisons.

Le synoptique de la commande MLI sinus triangle est le suivant :



- **Référence** : Placer une source de tension sinusoï dale *Vsin* paramétrée comme suit : VOFF = 0 VAMPL = 0.95V FREQ = {Fref}
- **Porteuse :** Placer une source de tension *Vpulse* paramétrée comme suit : V1 = -1V V2 = +1V TD = 0 TR = {0.5/Fporteuse} TF = {0.5/Fporteuse} PW = 1ns PER = {1/Fporteuse}
- **Paramètres :** Placer un composant *Param* avec les paramètres suivant : Name1 = Fref Name2 = Fporteuse Value1 = 50 Value2 = 1k
- **Comparateur :** Placer un composant *ABM2* de la bibliothèque ABM avec l'expression suivante : LIMIT((V(%IN1)-V(%IN2))*1e5,0,12)
- **Etiquettes :** Les entrées du comparateur seront étiquetées respectivement *Porteuse* et *Reference*.
- **Inverseur :** Placer un composant *ABM1* de la bibliothèque ABM avec l'expression suivante : 12-(V(%IN))
- **Liaison puissance-commande :** Placer sur la sortie du comparateur et celle de l'inverseur des composants *BUBBLE* que l'on nommera respectivement Q1 et Q2.

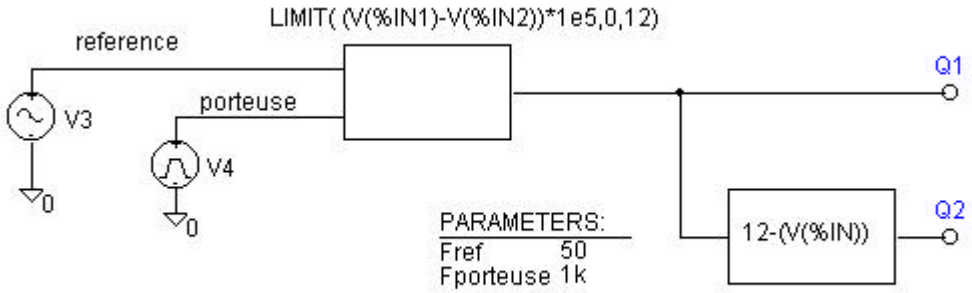
Remplacer les générateurs de tension *Vpulse* des grilles des IGBT par des composants *BUBBLE* nommés également Q1 et Q2.

Utilisation d'un module ABM pour créer un amplificateur opérationnel :
Les modules ABM (Analog Behavioral Modelling) permettent de mettre sous forme de composant des expressions mathématiques. Nous en avons déjà utilisé pour modéliser une MCC. Nous en utilisons à nouveau pour créer un modèle simplifié d'amplificateur opérationnel. On prend un module à deux entrées (*ABM2*) avec lequel on effectue successivement :

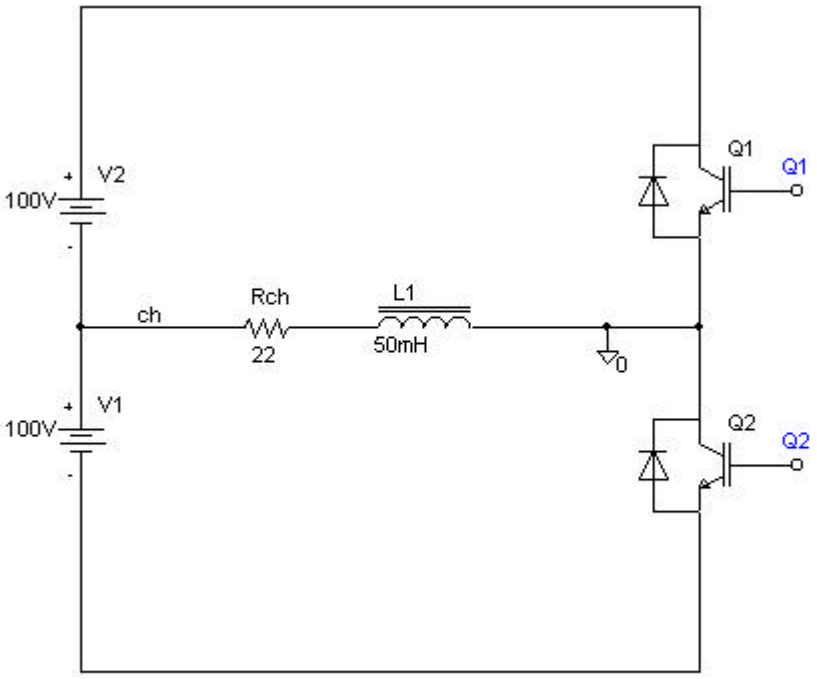
- La différence entre les deux entrées : V(%IN1)-V(%IN2)
- Puis le produit par un gain : (V(%IN1)-V(%IN2))*1e5
- Et enfin la limitation de la valeur de la tension de sortie : LIMIT((V(%IN1)-V(%IN2))*1e5,0,12)

On remarquera que la valeur inférieure de la saturation a été choisie dans ce cas précis à 0V plutôt que –12V dans un souci de simplification. Cette méthode a été employée pour créer le modèle simplifié d'amplificateur opérationnel disponible dans la bibliothèque ENREA.

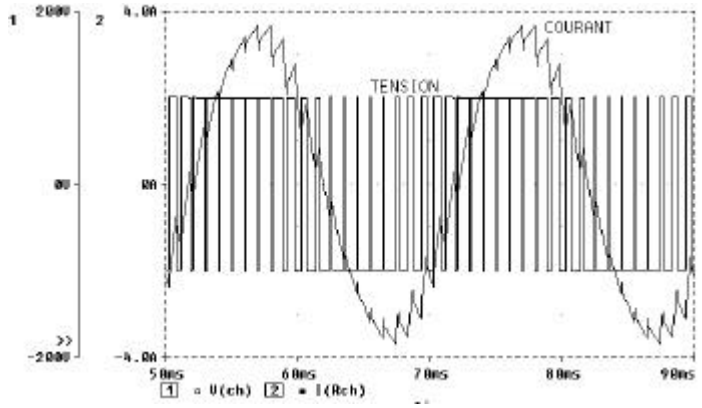
- Résultat attendu pour le circuit de commande :



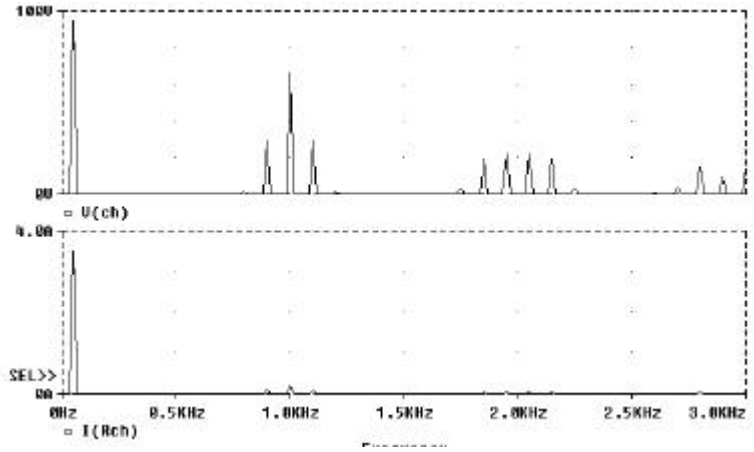
- Résultat attendu pour le circuit de puissance :



- Configurer une simulation temporelle de 60ms en supprimant le régime transitoire (*Final Time* = 100ms *No Print Delay* = 40ms). Lancer la simulation.
- On obtient :



- Analyse harmonique de la tension et du courant dans la charge :



On vérifie bien que les premiers harmoniques de tension sont regroupés autour de la fréquence de la porteuse. La charge étant inductive les harmoniques de courant correspondants sont très faibles. La valeur efficace du fondamental de la tension aux bornes de la charge est $V_1 = 66V$ ce qui est inférieur à la valeur obtenue en commande pleine onde.

THEME N°4 : SOURCE TRIPHASEE

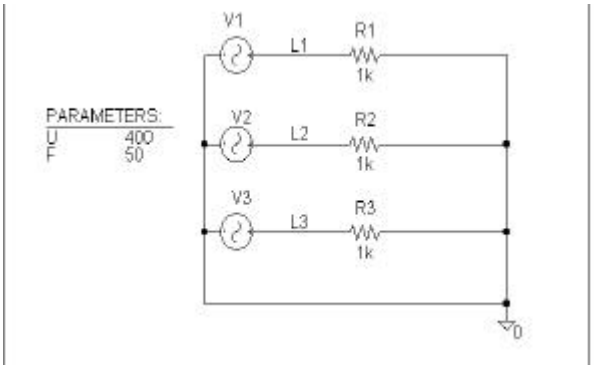
Présentation :

La version d'évaluation de Pspice autorise la configuration d'une bibliothèque personnelle contenant au maximum 20 composants. Nous allons traiter à partir d'un exemple la création de nouveaux composants. L'exemple choisi est une source de tension triphasée dont la tension et la fréquence sont paramétrables (valeur par défaut : 400V 50Hz). Nous l'appellerons *sourcetri.*
Nous modifierons deux fichiers : celui qui contient les symboles des composants, *enrea.slb* et celui qui contient les modèles des composants, *enrea.lib.*

Création d'un schéma :

Nous allons créer et essayer un schéma qui contient la fonction que l'on veut mettre sous forme de composant :

- Par un copier coller, placer le système triphasé du thème n°1 sur une nouvelle feuille. Modifier les paramètres définissant l'amplitude et la fréquence : VAMPL = {0.816*U}
  FREQ = {F}
- Placer un composant *param* qui permettra de passer en paramètre la tension U et la fréquence F de la source.
- Placer trois résistances de charge et la référence de potentiel.
- On obtient :

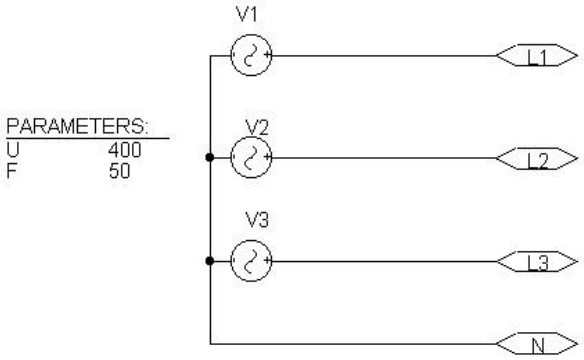


- Configurer une simulation temporelle avec les paramètres : *Print Step* = 0.1ms
  Final time = 60ms et vérifier le schéma.

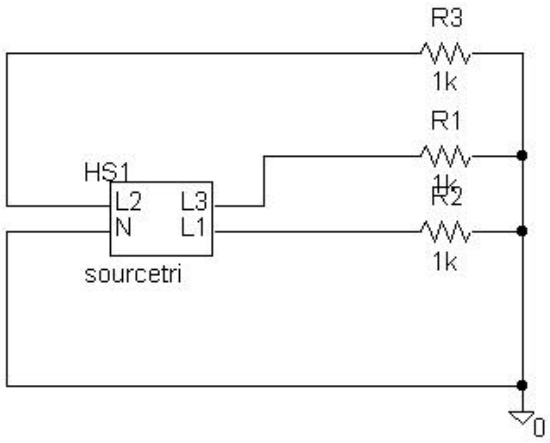
Création d'un sous-schéma :

Dans le schéma précédent :

- Supprimer ce qui a servi aux essais.
- Placer des composants *interface* sur les entrées-sorties.
- Nommer ces ports d'entrées-sorties.
- On obtient :



- Enregistrer le schéma sous le nom *sourcetri.sch*
- Faire *File Symbolise.* Donner le nom *sourcetri* au composant. Le sauver dans *enrea.slb.*
- Nous avons maintenant un composant supplémentaire nommé *sourcetri* dans notre bibliothèque *enrea.*
- Créer un nouveau schéma, placer un composant sourcetri, trois résistances, une référence de potentiel. On obtient :



- Faire l'essai.
- On peut en modifier les paramètres par un double-clic sur le symbole qui a pour effet d'ouvrir le schéma dont le symbole issu.

Création d'un sous-circuit :

Les inconvénients de l'utilisation d'un sous-schéma sont :

- Le sous-schéma doit être dans le même répertoire que le schéma où on veut l'utiliser.
- Lorsque l'on modifie un paramètre du sous-schéma, on le modifie pour toutes les occurrences de celui-ci.

Pour ces raisons, on préférera associer au symbole que l'on vient de créer un modèle qui sera ajouté à enrea.lib. Procéder comme suit :

- Ouvrir le schéma *sourcetri.sch.*
- Créer un sous-circuit avec *tools create subcircuit* (Pspice crée alors le fichier *sourcetri.sub* qui contient le modèle du composant).
- Ouvrir *sourcetri.sub* avec l'éditeur de texte de Pspice (aller dans Design Manager et faire *Tools TextEdit*) et ajouter les paramètres avec la syntaxe suivante :

```
.SUBCKT    sourcetri   L1  L2  L3  N   params: U=400V   F=50Hz
```

- Avec un copier-coller, ajouter le modèle ainsi modifié à *enrea.lib*.

Il faut maintenant associer un symbole à ce modèle :

- Ouvrir *Symbol Editor* par *File Edit library* , vous êtes dans l'éditeur de symbole de Pspice.
- Faire *Part Symbol Wizzard* .
- Choisir l'option *From a symbol library* puis cliquer sur *Next* .
- Sélectionner *enrea.lib* avec *Browse* puis *Ouvrir.*
- Sélectionner *enrea.slb* avec *Browse* puis *Ouvrir.*
- Choisr l'option *No, only create…*
- Sélectionner votre symbole dans l'une des deux listes puis cliquer sur *Generic Rectangle.*
- Ecrire une courte description de votre composant dans le cadre situé dessous puis cliquez sur *Finish* puis sur *Next* puis sur *Finish.*

La création de votre nouveau symbole est maintenant terminée, quitter l'éditeur de symboles par *File Close* et vérifiez le dans *Schematics.*

Modifier le symbole :

Pspice a donné un symbole par défaut à votre composant qui n'est guère satisfaisant, nous allons le modifier :

- Sélectionner le composant.
- Faire *Edit Symbol*.
- On se trouve alors dans l'utilitaire de création et de modification de symbole *Symbol Editor.*

La fonction dessin de *Symbol Editor* est tout à fait classique. Lorsque l'on édite un symbole il faut faire attention aux points suivants :

- Placer un cadre autour du symbole avec *Graphics Bbox.*
- Placer le point d'ancrage du symbole avec *Graphics Origin.*
- Les connexions sur les entrées-sorties se font par la petite croix.
- En faisant *Part Pin list* on peut modifier les caractéristiques des broches du composant : son nom, son numéro, ce qui est ou n'est pas affiché, son état si elle est « en l'air ».
- On choisit le résultat d'une broche « en l'air » en sélectionnant un choix avec *Float*= (*Error* = erreur *RtoGND* = reliée à la masse *UniqueNet* = sans effet).
- Si l'on souhaite modifier un attribut du circuit (faire apparaître la tension par exemple) faire *Part Attribute* puis éditer l'élément à modifier.
- Pour quitter *Symbol Editor* , il faut faire *File Close* et non *File Exit.*

Le nouveau symbole pourrait ressembler à ceci :